Music Genre Classification

Mark Gameng
Illinois Institute of Technology
mgameng1@hawk.iit.edu

1. Introduction

Music, throughout the years, have been skyrocketing in terms of music generation and have also resulted in new styles/genres of music. Music platforms in general are producing thousands of new songs each day, with Spotify having listed over 50 million songs and over 40 thousand new songs are added every day. Years ago, the genre of songs were classified manually by people with musical knowledge. With the amount of music generation in our current era, it is imperative that music genre classification be automated through musical analysis and machine learning. Automated music genre classification will allow platforms like Spotify, Pandora, Soundcloud to better serve its customers by having a more accurate search and recommender systems. Because of this, the interest in the field of Music Information Retrieval (MIR) has been increasing ever since. Specifically, one popular topic that has garnered countless studies in MIR is music genre classification. As a result, there has also been many datasets created containing songs with information like genre.

The datasets that are majorly used in these type of studies are GTZAN [8] and Extended Ballroom [1]. GTZAN is the most used public dataset which consists of 1000 audio clips that are 30 seconds each and are equally classified to 10 different genres. Meanwhile, the Extended Ballroom dataset consists of 4180 audio clips that are 30 seconds each which are from 13 different ballroom genres. Other datasets to note are Free Music Archive (FMA) and Million Song which are much larger datasets with 0.1 and 1 million clips respectively.

With machine learning, computer scientists have been trying to tackle the automation of music genre classification while using these datasets as a benchmark, and recently, have achieved pretty good results. A model can only learn to classify genre of music by looking at the features of it. Thus, feature extraction is one of the most important parts in music genre classification. In the past, feature extractions were done manually but currently, automatic feature extraction is done through machine learning. With just a spectrogram or any image the is able to accurately represent music, machine learning algorithms, more specifically Convolutional Neural Networks (CNN), are able to extract features and classify the genre at a very high accuracy.

Previous works have tried many ways to tackle music genre classification from logistic regression up to neural networks. Notably, most high accuracy models use neural networks either on its own, or combined with another algorithm, and they all achieve in the 80% range or higher. Specifically, a CNN [11], KCNN + SVM [10], and DNN [5] models got 87.4%, 83.9%, and 83% respectively. The best model that achieved the highest accuracy is done using a parallel recurrent convolutional neural network (PRCNN) [9] which got 92% in GTZAN and 92.5% on Extended Ballroom dataset.

Most of the methods used above are trained on only one type of spectrogram, Short-time Fourier Transforms (STFT). While STFTs alone may be enough to achieve a good genre classification accuracy, combining models with different spectogram inputs may result in higher accuracies. This combination would be somewhat similar to the architecture proposed by Yang et al. [9] that used a CNN and a bi-directional RNN in parallel with an input of STFT. Lower end models such as logistic regression, support vector clustering/machine will probably also increase accuracy when combined with other higher end models. These are the things I will be looking at and experimenting on.

2. Data

GTZAN [8] dataset is widely used for music genre classification. It is a public dataset that consists of 1,000 audio clips that are 30 seconds each and are equally classified to 10 different genres. The 10 genres are blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, and rock. There have been papers indicating the faults of the dataset, such as by Sturm [6]. Sturm informs that while GTZAN has its faults, it can still be used for music genre classication tasks if used with consideration of its faults.

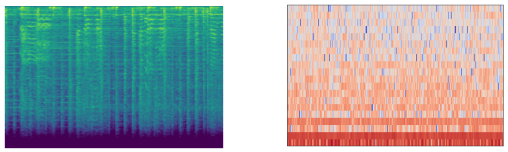
Extended Ballroom [1] is another dataset that is gaining popularity which is an improved version of the well-known Ballroom dataset. This dataset contains 4,180 tracks that are 30 seconds each and are classified to 13 different ballroom genres. The 13 genres are chacha, jive, quick-

step, rumba, samba, tango, viennesewaltz, waltz, foxtrot, pasodoble, salsa, slowwaltz, and wcswing, Out of the 13, 10 genres have 252 to 529 tracks each while the last 3 genres only have about 50 tracks each.

2.1. Processing

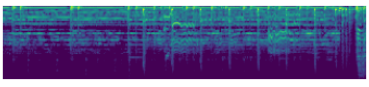
GTZAN and Extended Ballroom only has 1,000 and 4,180 tracks with 10 and 13 genres respectively. GTZAN has equally distributed tracks while Extended Ballroom does not. Thus, 4 ballroom genres- Pasodoble, Salsa, Slow Waltz, Wcswing - were removed for having less than 100 tracks. This resulted in 1,000 tracks for GTZAN and 3,992 tracks for Extended Ballroom. The tracks were 30 seconds each and thus are able to be split up into shorter segments. This would allow for more data to train and test on. I opted to splitting the tracks by 10, 8, 5, 4, 2, 1 segments subsequently, which would result in more samples while also being able to test how much the time affects accuracy for music genre classification. This resulted in at max, 10,000 samples for GTZAN and 39,920 samples for Extended Ballroom. Compared to the original dataset, this amount of samples is much better and would result in the models less likely to overfit.

The following features were extracted using the Python Librosa package: Short-time Fourier Transform (STFT), Mel-frequency Cepstral Coefficients (MFCC), and Mel-spectrogram, which are also shown in Figure 1. Currently, STFT is the most widely used in the top accuracy models done by others. STFT simply represents a signals amplitude as it varies over time at different frequencies. MFCC are coefficients that make up an Mel-frequency spectrum (MFC) that has the frequency bands equally spaced on the mel scale which is similar to how humans perceive sounds. Somewhat similar to MFCCs, Melspectrograms are spectograms where the frequencies are converted to mel scale.



(a) STFT

(b) MFCC



(c) Melspectrogram

Figure 1: Feature Extraction from Audio

3. Methodology

Due to the low amount of samples in both GTZAN and Extended Ballroom dataset, splitting the samples into multiple segments would be beneficial in terms of accuracy and generalization. The tracks were split into 10, 8, 5, 4, 2, 1 segments with no overlap. Thus the samples ranged from 3 seconds to 30 seconds of audio. With a split of 10 segments for each sample, GTZAN now has 10,000 samples and Extended Ballroom with 39,920.

For model training, 20% of samples were set aside for testing. This resulted in 80% of the samples being used for training, in which 20% of that is used for validation. Thus, it is a 64-16-20 split for training, validation, and testing. Aside from sample splitting, other methods were used such as early stopping, regularization, and dropouts for neural networks to avoid overfitting. While training, the best model is determined and used using the least validation loss through model checkpoints. The following sections describe the methods and architectures used. Currently, I have only done CNN as I have yet to do RNN, SVM and other stuff.

3.1. CNN

Due to the different shapes of the spectrograms for the input, the models for each feature vary a little from each other, however the general structure is the same. The spectograms has the shapes (1025, 64), (20, 64), (64, 64) with 10 splits for STFT, MFCC, and Melspectrogram respectively. As a result for the smaller shapes, I decided to use 4 convolutional layers for MFCC and Melspectrogram while using 5 for STFT. After each convolutional layer, a max pooling operation and dropout is done to overall reduce overfitting and extract only the most needed music features. The convolutional layers all have padding, kernel size of (3, 3), and strides of (1, 1) with rectified linear units (ReLUs) activation function. The filters for STFT are (16, 32, 64, 128, 64) and for MFCC and Melspectrogram, (32, 64, 128, 64). For the max pooling operation, pool size is (2, 2) and strides is (2, 2). For each convolutional layer the dropout is 0.25.

After it has gone through all the convolutional layers, the output is flattened and a 0.5 dropout is once again used. The output is then fed to a dense layer of 256 neurons and a ReLU activation with l2 regularization. Afterwards, a 0.25 dropout is used, and then the final dense layer of #genres neurons with softmax activation to classify the music genres.

3.2. To Do

I have yet to form and test RNN architectures and combining models together to achieve higher accuracies. RNN would be useful and I think more successful than CNN because they can extract temporal information from the spectrogram. For the parallel/combination of models, I will be

trying same input as well as different feature inputs and see which works best. I may also combine using classical methods such as logistic regression, random forest, SVMs, etc. After optimizing, I will also be using majority voting since the clips were split into segments and seeing if it results in better accuracies.

4. Results

The models specified in the previous section were trained using the GTZAN and Extended Ballroom dataset and the results from each of the different models are shown in Table 1 and 2, for GTZAN and Extended Ballroom respectively. Only the top models are shown, with the accuracy of other models included for comparison. Due to utilizing different features and optimization, the general structure of the models are similar, but not one to one. Only the best models of each type are thoroughly described in the methodology section.

Table 1: Results on GTZAN

Model	Feature	Accuracy
CNN + 1-layer RNN [9]	STFT	90.2%
CNN + 2-layer RNN [9]	STFT	88.8%
CNN [9]	STFT	88.0%
CNN - 10 split	**STFT**	**86.2%**
CNN - 4 split	**Melspectrogram**	**83.1%**
2-layer BRNN [4]	STFT	76.2%

In terms of the GTZAN dataset, the models described in the previous section resulted in respectable accuracies compared to others. A simple 5 and 4 layer CNN model using STFT and Melspectogram achieved 86.2% and 83.1% respectively. Those models are what I currently have done and I have not even tried optimizing them yet. After optimizing them, I will be combining them with each other and other models, similar to the model made by Yang et al. [9], however with different feature inputs. The resulting models will be placed in Table 1 as well.

Table 2: Results on Extended Ballroom

Model	Feature	Accuracy
CNN + 1-layer RNN [9]	STFT	92.3%
CNN + 2-layer RNN [9]	STFT	92.5%
CNN [9]	STFT	92.2%
2-layer BRNN [4]	STFT	90.3%
CNN - 4 split	**Melspectrogram**	**89.9%**
CNN - 10 split	**STFT**	**88.5%**
CNN - 8 split	**MFCC**	**82.3%**

In terms of the Extended Ballroom dataset, the models described in the previous section resulted in respectable but lower accuracies compared to other models tested in the same dataset. I haven't spent time optimizing the models, so the accuracies might still increase and be more similar or better than other models. I will also be combining them together or with other models to try and achieve even better accuracies. The resulting models will be placed in Table 2 as well.

Currently, I have only trained CNN models that I think results in the best accuracy. Thus, the results don't include all the possible models with splits and features. However, the current results are already promising, with the models having at least 80% accuracy. With optimization and added features such as majority voting or using models together, the accuracy can still be increased. I also intend on training the models multiple times to instead show the average accuracy rate, which, I think, is more representative.

Using multiple splits to train will be helpful in determining up to what length of audio can a model accurate classify the genre. Doing so will allow me to choose the best model based on effectiveness and efficiency. The results for each split and models are shown in Table 3 and 4, for GTZAN and Extended Ballroom respectively.

Table 3: Impact of Splits in GTZAN

Model	Feature	Split	Accuracy
CNN	STFT	10	86.2%
CNN	STFT	8	86.4%
CNN	STFT	4	81.5%
CNN	STFT	1	57.0%
CNN	MFCC	1	TBD%
CNN	MFCC	4	TBD%
CNN	MFCC	8	73.8%
CNN	MFCC	10	TBD%
CNN	Melspectrogram	4	83.1%
CNN	Melspectrogram	2	TBD%
CNN	Melspectrogram	1	TBD%
CNN	Melspectrogram	10	TBD%

References

[1] Ugo Marchand and Geoffroy Peeters. The extended ballroom dataset. 2016.

[2] Dirceu de Freitas Piedade Melo, Inacio de Sousa Fadigas, and Hernane Borges de Barros Pereira. Graph-based feature extraction: A new proposal to study the classification of music signals outside the time-frequency domain. *Plos one*, 15(11):e0240915, 2020.

[3] Sergio Oramas, Oriol Nieto, Francesco Barbieri, and Xavier Serra. Multi-label music genre classification from au-

Table 4: Impact of Splits in Extended Ballroom

Model	Feature	Split	Accuracy
CNN	STFT	10	88.5%
CNN	STFT	8	TBD%
CNN	STFT	4	TBD%
CNN	STFT	1	TBD%
CNN	MFCC	10	TBD%
CNN	MFCC	8	82.3%
CNN	MFCC	4	TBD%
CNN	MFCC	1	TBD%
CNN	Melspectrogram	4	89.9%
CNN	Melspectrogram	2	TBD%
CNN	Melspectrogram	1	TBD%
CNN	Melspectrogram	10	TBD%

dio, text, and images using deep features. *arXiv preprint arXiv:1707.04916*, 2017.

[4] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

[5] Siddharth Sigtia and Simon Dixon. Improved music feature learning with deep neural networks. In *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 6959–6963. IEEE, 2014.

[6] Bob L Sturm. The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use. *arXiv preprint arXiv:1306.1461*, 2013.

[7] Alexandros Tsaptsinos. Lyrics-based music genre classification using a hierarchical attention network. *arXiv preprint arXiv:1707.04678*, 2017.

[8] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on speech and audio processing*, 10(5):293–302, 2002.

[9] Rui Yang, Lin Feng, Huibing Wang, Jianing Yao, and Sen Luo. Parallel recurrent convolutional neural networks-based music genre classification method for mobile devices. *IEEE Access*, 8:19629–19637, 2020.

[10] Pengjing Zhang, Xiaoqing Zheng, Wenqiang Zhang, Siyan Li, Sheng Qian, Wenqi He, Shangtong Zhang, and Ziyuan Wang. A deep neural network for modeling music. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pages 379–386, 2015.

[11] Weibin Zhang, Wenkang Lei, Xiangmin Xu, and Xiaofeng Xing. Improved music genre classification with convolutional neural networks. In *Interspeech*, pages 3304–3308, 2016.